REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application. Claims 1-41 are pending in this application.

35 U.S.C. § 103

Claims 1-3, 4, 6-17, 19-21, 23-27, 29-38, and 40-41 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,138,271 to Keeley (hereinafter "Keeley") in view of U.S. Patent No. 5,325,533 to McInerney et al. (hereinafter "McInerney"). Applicant respectfully submits that claims 1-3, 4, 6-17, 19-21, 23-27, 29-38, and 40-41 are not obvious over Keeley in view of McInerney.

Keeley is directed to a computer operating system adapted for use in a limited memory computer that may be embedded in other products (see, col. 1, lines 6-8). More specifically, as discussed in the Abstract of Keeley, Keeley is directed to a software development system including a modular operating system program having a plurality of modules each providing an operating system operation that may be called by an application program. The operating system operations permit application programs to communicate with particular computer hardware or other executing programs. A scanner program reviews the application program for calls to operating system operations to produce an OS module list of such application calls; and a selective compiler program receiving the OS module list prepares an operating system comprised of only those modules of the modular operating system necessary to perform the application calls of the OS module list that may be loaded into the embedded computer.

McInerney is directed to a human oriented object programming system (HOOPS) which provides an interactive and dynamic environment for computer program building (see, col. 1, lines 9-11). Program building is made possible by the interaction of an incremental program model, called a project, and three major functionalities (see, col. 3, lines 4-7). A program is modeled as semantic units called components, each component representing a single compilable language element such as a class or a function (see, col. 3, lines 7-8 and 15-17). The three major functionalities are the database, the compiler and the build mechanism (see, col. 3, lines 25-26). The database persistently stores and retrieves the components and their properties (see, col. 3, lines 26-28). The compiler, along with compiling the source code of a property, is responsible for calculating the dependencies associated with a component (see, col. 3, lines 28-30). The build mechanism uses properties of components along with the compiler generated dependencies to correctly and efficiently sequence the compilation of components during a build process (see, col. 3, lines 30-34).

As discussed at MPEP §§ 2142 and 2143, to establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be

found in the prior art, and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Applicant respectfully submits that Keeley and McInerney do not teach or suggest all of the elements of the claims, and thus that no *prima facie* case of obviousness has been established.

With respect to claim 1, claim 1 recites:

> A computer-implemented method, comprising:
> creating a software development kit object (SDK object) for at least some of a plurality of development files in a source operating system that includes development files and components;
> identifying features of the source operating system to be included in a modularized system that is a subset of the source operating system;
> tracing dependencies in a dependency model correlating to the source operating system that uses the SDK objects to identify SDK objects corresponding to development files that are required to support the identified features;
> selecting the development files that correspond to the identified SDK objects; and
> exporting the selected development files to a software development kit (SDK) that supports development of applications for use with the modularized system.

Applicant respectfully submits that Keeley in view of McInerney does not disclose or suggest the method of claim 1.

In the December 27, 2005 Final Office Action at p. 6, Keeley at col. 3, lines 10-11 is cited as disclosing the selecting of claim 1. Applicant respectfully disagrees and submits that this cited portion of Keeley does not disclose or suggest the selecting of claim 1. Keeley at col. 3, lines 9-13 reads:

> Thus, it is one object of the invention to provide an operating system that is modular so that selected modules may be collected and reformed into a smaller operating system necessary for a particular application program in an embedded environment.

Thus, this cited portion of Keeley discusses selecting modules to be reformed into an operating system, not selecting development files. Although Keeley refers to a development system (see, col. 2, lines 46-49), there is no discussion or mention in the cited portion of Keeley, or elsewhere in Keeley, of selecting development files that correspond to identified SDK objects. Without any such discussion or mention, Applicant respectfully submits that Keeley cannot disclose or suggest selecting the development files that correspond to the identified SDK objects as recited in claim 1.

In the December 27, 2005 Office Action at pp. 3-4, it was asserted that:

> In response to Applicant's argument, Keeley's system provides a development system that allows application programs to be developed in the environment of a complex operating system. Where his system select the only modules (development files) those are needed to build the particular operating system (See the Summary). Thus, the system does provide selecting the development files for an identified embedded operating system. Applicant only makes general allegations. Therefore, the rejection is proper and maintained herein.

Applicant respectfully disagrees with this characterization of Keeley. The modules discussed in the Summary of Keeley are modules that the operating system is partitioned into (see, col. 2, lines 50-52, and col. 3, lines 9-13). These modules of Keeley are not development files that correspond to identified software development kit objects. As discussed in Applicant's specification at, for example, p. 21, lines 1-3, the development files may include header files, library files, documentation files, auxiliary files and the like, that are required to utilize the target operating system and to create applications to run in conjunction with the target operating system. The different modules that the operating system is partitioned into in Keeley are not such development files as recited in claim 1.

Accordingly, Applicant respectfully submits that Keeley does not disclose or suggest selecting the development files that correspond to the identified SDK objects as recited in claim 1.

Additionally, in the December 27, 2005 Final Office Action at p. 6, Keeley at col. 3, lines 4-7 is cited as disclosing the exporting of claim 1. Applicant respectfully disagrees and submits that this cited portion of Keeley does not disclose or suggest the exporting of claim 1. Keeley at col. 3, lines 1-8 reads:

> A scanner program reviews the application program for calls to operating system operations to produce an OS module list of such application calls; and a selective compiler program receiving the OS module list prepares an operating system comprised of only those modules of the modular operating system necessary to perform the application calls of the OS module list that may be loaded into the embedded computer.

Thus, this cited portion of Keeley is directed to preparing an operating system comprised of only those modules of the operating system necessary to perform the application calls. Thus, it can be seen that Keeley is discussing preparing an operating system, not preparing a software development kit. Although Keeley refers to a development system (see, col. 2, lines 46-49), there is no discussion or mention in the cited portion of Keeley, or elsewhere in Keeley, of exporting selected development files to a software development kit. Without any such discussion or mention, Applicant respectfully submits that Keeley cannot disclose or suggest exporting the selected development files to a software development kit (SDK) that supports development of applications for use with the modularized system as recited in claim 1.

In the December 27, 2005 Office Action at p. 4, it was asserted that:

In response to Applicant's argument, Keeley's does not mention exporting selected development files to a software development kit. Keeley does disclose in FIG. 4 developing of application program, where the system uses a scanner to collect (export) a list of operation names used in the application program by comparison to those names to the API (col. 6, lines 17-67 and col. 7-8). Thus, the system does provide exporting feature as claimed in claim 1. Therefore, the rejection is proper and maintained herein.

From this assertion in the December 27, 2005 Office Action, it appears that the office has acknowledged that Keeley does not mention exporting the selected development files to a software development kit as recited in claim 1. Without any such mention, Applicant respectfully submits that Keeley cannot disclose exporting the selected development files to a software development kit (SDK) that supports development of applications for use with the modularized system as recited in claim 1.

Furthermore, in Fig. 4 of Keeley an application program for an embedded application is developed with a full-featured operating system that is constructed in modular form (see, col. 6, lines 18-21). The application program is reviewed by a scanner program which reads the text of the application program to collect a list of operation names used in the application program by comparison of those names to the API (see, col. 6, lines 59-64). The scanner collects each operating system operation name in an OS module list (see, col. 7, lines 6-7). The OS module list is used by a builder program which also reviews an OS cross reference table and determines what other modules are called by the operations detected in the application source code (see, col. 7, lines 51-54). In turn, each of these modules is examined (via the cross reference table) for calls it makes to other operating system modules until no new called modules are found and a complete set of operations necessary to run the application source has been determined (see,

col. 7, lines 54-59). The builder selects from the operating system all the operations directly or indirectly referenced by the application program as identified by the scanner (in the OS module list) and by the OS cross reference table and produces a reduced OS containing a subset of the source code of the operating system (see, col. 7, line 64-col. 8, line 2).

Thus, it can be seen that Keeley discusses collecting operating system operation names and selecting all the operations directly or indirectly referenced by the application program. However, these are operating system operations and operation names used to produce a reduced OS containing a subset of the source code of the operating system, not development files exported to a software development kit. Nowhere in this discussion of Keeley is there any mention or discussion of exporting selected **development files to a software development kit** that supports development of applications for use with a modularized system. Accordingly, Applicant respectfully submits that Keeley does not disclose or suggest exporting the selected development files to a software development kit (SDK) that supports development of applications for use with the modularized system as recited in claim 1.

In addition, Keeley is cited in the December 27, 2005 Office Action at pp. 5-6 as disclosing the creating, identifying, selecting, and exporting of claim 1. Keeley states that "The present invention provides a development system that allows application programs to be developed **in the environment of a complex operating system and then to be loaded into embedded computers having limited memory**" (see, emphasis added, col. 2, lines 46-49). Keeley also discloses that development of an application program may be done on a standard

desk-top computer system using the entire modular operating system installed or an equivalent non-modular operating system (see, col. 9, lines 9-13). Additionally, one of the objects of Keeley is to permit programmers developing applications for embedded computers to develop their application programs using a full-featured operating system that supports high level languages and development tools (see, col. 3, lines 14-18).

In contrast, claim 1 recites exporting the selected development files to a software development kit (SDK) that supports development of applications for use with a modularized system, the selected development files corresponding to identified SDK objects that in turn correspond to development files that are required to support features of the source operating system identified to be included in the modularized system that is a subset of the source operating system. If the goal of Keeley is to permit programmers to develop their applications using a full-featured operating system, it would be nonsensical to perform any of the identifying, selecting or exporting as recited in claim 1 in Keeley – Keeley is using a full-featured operating system for development so there would be no need to differentiate between development files that are required to support the identified features and those that are not required. As such, Applicant respectfully submits that Keeley does not disclose or suggest the identifying, selecting, and exporting as recited in claim 1.

Applicant notes that although this argument was raised in the Response filed October 5, 2005, this argument was not addressed in the December 27, 2005 Final Office Action. If the rejection of claim 1 is maintained, Applicant respectfully requests that this argument be addressed.

With respect to McInerney, Applicant respectfully submits that McInerney is not cited as curing, and does not cure, these deficiencies of Keeley.

For at least these reasons, Applicant respectfully submits that claim 1 is allowable over Keeley in view of McInerney.

With respect to claims 2-4, 6, 7, and 10-12, given that claims 2-4, 6, 7, and 10-12 depend from claim 1, Applicant respectfully submits that claims 2-4, 6, 7, and 10-12 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 1.

With respect to claim 8, claim 8 depends from claim 1 and Applicant respectfully submits that claim 8 is allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 1. Furthermore, claim 8 recites:

> The method as recited in claim 1, wherein creating the SDK objects further comprises:
> naming a data object having a type that identifies the data object as being an SDK object;
> including at least one reference in a first SDK object, the reference pointing to a second SDK object that is required by the first SDK object to function properly; and
> repeating the previous steps for each development file to be exposed in the SDK.

Applicant respectfully submits that no software development kit object (SDK object) is disclosed in Keeley, much less naming a data object having a type that identifies the data object as being an SDK object as recited in claim 1.

In the December 27, 2005 Office Action at p. 11, Keeley at col. 5, lines 6-7 is cited as disclosing the naming of claim 8. Applicant respectfully disagrees and submits that this cited portion of Keeley does not disclose the naming of claim 8. The sentence including this cited portion of Keeley reads:

> The application program calls the operating system operations via an application programmer's interface ("API") 28 which provides a standard set of operation names 40 that are linked to particular modules 44 of the operating system holding corresponding operations, each operation being a routine for performing the desired operation (e.g., a disk read, etc.).

Thus, it can be seen that Keeley is discussing an application programmer's interface and operation names provided by such an interface. There is no reference at all to a software development kit object in this portion of Keeley.

Without even so much as a mention of a software development kit object, Applicant respectfully submits that Keeley cannot disclose or suggest naming a data object having a type that identifies the data object as being an SDK object as recited in claim 8.

Applicant notes that although this argument regarding claim 8 was raised in the Response filed October 5, 2005, this argument was not addressed in the December 27, 2005 Final Office Action. If the rejection of claim 8 is maintained, Applicant respectfully requests that this argument be addressed.

For at least these reasons, Applicant respectfully submits that claim 8 is allowable over Keeley in view of McInerney.

With respect to claim 9, given that claim 9 depends from claim 8, Applicant respectfully submits that claim 9 is likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 8.

With respect to claim 13, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the selecting and exporting of claim 13. For at least these

reasons, Applicant respectfully submits that claim 13 is allowable over Keeley in view of McInerney.

With respect to claims 14-17, given that claims 14-17 depend from claim 13, Applicant respectfully submits that claims 14-17 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 13.

With respect to claim 19, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the identifying, selecting, and exporting of claim 19. For at least these reasons, Applicant respectfully submits that claim 19 is allowable over Keeley in view of McInerney.

With respect to claims 20-21 and 23-24, given that claims 20-21 and 23-34 depend from claim 19, Applicant respectfully submits that claims 20-21 and 23-34 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 19.

With respect to claim 25, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the selecting and filtering of claim 25. For at least these reasons, Applicant respectfully submits that claim 25 is allowable over Keeley in view of McInerney.

With respect to claim 26, claim 26 depends from claim 25 and Applicant respectfully submits that claim 26 is allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 25. Furthermore, claim 26 recites:

The one or more computer-readable media as recited in claim 25, wherein the filtering further comprises:

for each selected SDK object, searching for a label in the master SDK header file that is the same name as the SDK object;

if the label is found, enabling a section of code associated with the label; and

wherein the enabling the section of code associated with the label enables the appropriate development file associated with the SDK object to be included in the SDK.

Applicant respectfully submits that no SDK object is disclosed in Keeley, much less for each selected SDK object, searching for a label in the master SDK header file that is the same name as the SDK object as recited in claim 26.

Applicant respectfully submits that there is no discussion or mention of a master SDK header file in Keeley or McInerney, and that there is no portion of either Keeley or McInerney cited in the December 27, 2005 Office Action as disclosing a master SDK header file. Without such a discussion or mention, Applicant respectfully submits that Keeley and McInerney cannot disclose or suggest searching for a label in a master SDK header file as recited in claim 26.

Furthermore, similar to the discussion above regarding claim 8, Applicant respectfully submits that there is no discussion or mention in Keeley or McInerney of a name of an SDK object as recited in claim 26. Without such a discussion or mention, Applicant respectfully submits that Keeley and McInerney cannot disclose or suggest searching for a label in the master SDK header file that is the same name as the SDK object as recited in claim 26.

Applicant notes that although these arguments regarding claim 26 were raised in the Response filed October 5, 2005, these argument was not addressed in the December 27, 2005 Final Office Action. If the rejection of claim 26 is maintained, Applicant respectfully requests that these arguments be addressed.

For at least these reasons, Applicant respectfully submits that claim 26 is allowable over Keeley in view of McInerney.

With respect to claim 27, claim 27 depends from claim 25 and Applicant respectfully submits that claim 27 is allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 25. Furthermore, Applicant respectfully submits that, similar to the discussion of claim 26 above, Keeley in view of McInerney does not disclose or suggest the executing and determining of claim 27. For at least these reasons, Applicant respectfully submits that claim 27 is allowable over Keeley in view of McInerney.

With respect to claims 29-33, given that claims 29-33 depend from claim 25, Applicant respectfully submits that claims 29-33 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 25.

With respect to claim 34, Applicant respectfully submits that, similar to the discussion above regarding claim 1, Keeley in view of McInerney does not disclose or suggest the SDK object generator, the feature identification module, the dependency tracer, and the export module of claim 34. For at least these reasons, Applicant respectfully submits that claim 34 is allowable over Keeley in view of McInerney.

With respect to claims 35-38 and 40-41, given that claims 35-38 and 40-41 depend from claim 34, Applicant respectfully submits that claims 35-38 and 40-41 are likewise allowable over Keeley in view of McInerney for at least the reasons discussed above with respect to claim 34.

Claims 5, 18, 22, 28, and 39 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Keeley in view of U.S. Patent No. 5,901,319 to Hirst (hereinafter "Hirst"). It appears from the December 27, 2005 Office Action that McInerney is also relied on in the rejection of claims 5, 18, 22, 28, and 39 (see, December 27, 2005 Office Action at ¶ 10, p. 18). Applicant respectfully submits that claims 5, 18, 22, 28, and 39 are not obvious over Keeley in view of Hirst.

Claims 5, 18, 22, 28, and 39 depend from claims 1, 13, 19, 25, and 34, respectively. Applicant respectfully submits that claims 5, 18, 22, 28, and 39 are allowable over Keeley in view of McInerney at least because of their dependency on claims 1, 13, 19, 25, and 34, respectively. Applicant further submits that Hirst is not cited as curing, and does not cure, the deficiencies of Keeley in view of McInerney discussed above. For at least these reasons, Applicant respectfully submits that claims 5, 18, 22, 28, and 39 are allowable over Keeley in view of Hirst, as well as Keeley in view of McInerney and Hirst.

Applicant respectfully requests that the §103 rejections be withdrawn.

Conclusion

Claims 1-41 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. Should any matter in this case remain unresolved, the undersigned attorney respectfully requests a telephone conference with the Examiner to resolve any such outstanding matter.

Respectfully Submitted,

Date: 2/22/06   By: _____

Allan T. Sponseller
Reg. No. 38,318
(509) 324-9256